# On Sensor Network Localization Using SDP Relaxation

Davood Shamsi*      Nicole Taheri†      Zhisu Zhu‡      Yinyu Ye*

January 4, 2012

## Abstract

A Semidefinite Programming (SDP) relaxation is an effective computational method to solve a Sensor Network Localization problem, which attempts to determine the locations of a group of sensors given the distances between some of them [11]. In this paper, we analyze and determine new sufficient conditions and formulations that guarantee that the SDP relaxation is exact, i.e., gives the correct solution. These conditions can be useful for designing sensor networks and managing connectivities in practice.

Our main contribution is twofold: We present the first non-asymptotic bound on the connectivity or radio range requirement of the sensors in order to ensure the network is uniquely localizable. Determining this range is a key component in the design of sensor networks, and we provide a result that leads to a correct localization of each sensor, for any number of sensors. Second, we introduce a new class of graphs that can always be correctly localized by an SDP relaxation. Specifically, we show that adding a simple objective function to the SDP relaxation model will ensure that the solution is correct when applied to a triangulation graph. Since triangulation graphs are very sparse, this is informationally efficient, requiring an almost minimal amount of distance information. We also analyze a number objective functions for the SDP relaxation to solve the localization problem for a general graph.

## 1 Introduction

Graph Realization is a commonly studied topic which attempts to map the nodes in a graph $G(V, E)$ to point locations in the Euclidean space based on the non-negative weights of the edges in $E$; that is, the weight of each edge corresponds to the Euclidean distance between the incident points. There are a number of applications of the graph realization problem [9, 13, 16, 21, 25]. In this paper, we focus on the application to Sensor Network Localization (SNL).

A sensor network consists of a collection of *sensors* whose locations are unknown, and *anchors* whose locations are known. A common property of a sensor network is that each sensor detects others within a given connectivity (or radio) range and determines the distance from itself to these other sensors. Given this set of sensors and distances, the goal is to determine the exact location of each of the sensors. The problem becomes a graph realization problem by forming the weighted

---

*Department of Management Science and Engineering, Stanford University, Stanford, CA 94305. Email: davood@stanford.edu, , yinyu-ye@stanford.edu

†Institute for Computational and Mathematical Engineering, Stanford University, Stanford, CA 94305. Email: ntaheri@stanford.edu

‡Oracle Corporation, 500 Oracle Parkway, Redwood Shores, California 94065. Email: zhuzhisu@gmail.com

undirected graph $G(V, E)$, where the node set $V$ represents the sensors and each non-negative weighted edge in $E$ represents a known distance between two sensors.

The SNL problem has received a lot of attention recently because of the formulation of its relaxation as a Semidefinite Program (SDP) [2, 11, 24, 30]. This formulation can find the exact locations of the sensors, given that the graph possesses certain properties. In this paper, present a number of additional sufficient conditions that guarantee unique localizability of the SDP relaxation of the SNL problem, i.e., conditions that ensure the SDP will give the correct solution in the space of a desired dimension. These conditions would be useful for designing sensor networks and managing connectivities in practice.

## 1.1 Background

We are given a graph $G(V, E \cup \bar{E})$ in a fixed dimension $d$, where the nodes or points of $V$ are partitioned into two sets: the set $V_a = \{a_1, \ldots, a_m\}$ of $m$ anchors (where $m \geq d + 1$) whose locations are known and the set $V_x = \{x_1, \ldots, x_n\}$ of $n$ sensors, whose locations are unknown. The edge set also consists of two distinct sets, $E$ and $\bar{E}$, where $E = \{(i, j) : i, j \in V_x\}$ is the set of edges between sensors, and $\bar{E} = \{(k, j) : k \in V_a, j \in V_x\}$ is the set of edges between an anchor and a sensor. Moreover, for each $(i, j) \in E$ (or $(k, j) \in \bar{E}$) the distance value between sensor $i$ and sensor $j$ (resp. anchor $k$ and sensor $j$) is known as $d_{ij}$ (resp. $\bar{d}_{kj}$). The problem of finding the locations of the sensors can be formulated as finding points $x_1, x_2, \ldots, x_n \in \mathbb{R}^d$ that satisfy a set of quadratic equations:

$$
\begin{aligned}
\| x_i - x_j \|^2 &= d_{ij}^2, \quad \forall\, (i, j) \in E \\
\| a_k - x_j \|^2 &= \bar{d}_{kj}^2, \quad \forall\, (k, j) \in \bar{E}.
\end{aligned}
\tag{1}
$$

From this, a number of fundamental questions naturally arise: Is there a localization or realization of $x_j$'s that solves this system? If there is a solution, is it unique? And is there a way to certify that a solution is unique? Is the network instance partially localizable, i.e., is the localization solution for a subset of the sensors unique? These questions were extensively studied in the graph rigidity and discrete geometry communities from a more combinatorial and theoretical prospective (see [15, 20] and references therein). However, the computational aspects of these questions remained open, that is, the question of whether there is an efficient algorithm to numerically answer some of these questions was left open.

The SDP relaxation model (2) and corresponding method aim to answer these questions computationally (see [11, 30]). For $e_i \in \mathbb{R}^n$, the $i$th column of the identity matrix in $\mathbb{R}^{n \times n}$, we define symmetric coefficient matrices $A_{ij} := (\mathbf{0}; e_i - e_j)(\mathbf{0}; e_i - e_j)^T$ and $\bar{A}_{kj} := (a_k; -e_j)(a_k; -e_j)^T$, where $\mathbf{0} \in \mathbb{R}^d$ is the vector of all zeros. The SDP relaxation can be represented as:

$$
\begin{aligned}
\text{maximize} \quad & 0 \\
\text{subject to} \quad & Z_{(1:d, 1:d)} = I_d \\
& A_{ij} \bullet Z = d_{ij}^2, \quad \forall (i, j) \in E \\
& \bar{A}_{kj} \bullet Z = \bar{d}_{kj}^2, \quad \forall (k, j) \in \bar{E} \\
& Z \succeq 0.
\end{aligned}
\tag{2}
$$

Here, $Z_{(1:d, 1:d)}$ represents the upper-left $d$-dimensional principle submatrix of $Z$, the matrix dot-product refers to the sum of element-wise products $A \bullet B = \sum_{ij} A_{ij} B_{ij}$, and $Z \succeq 0$ means that the symmetric variable matrix $Z$ is positive semidefinite. Note that problem (2) is a convex semidefinite program and can be approximately solved in polynomial time by interior-point algorithms.

2

One can see that the solution matrix $Z \in \mathbb{R}^{(d+n)\times(d+n)}$ of (2) is a matrix that can be decomposed into submatrices,

$$Z = \begin{bmatrix} I & X \\ X^T & Y \end{bmatrix}.$$

The constraint $Z \succeq 0$ implies that $Y \succeq X^T X$. If $Y = X^T X$, then the above formulation finds a matrix $Z$ such that the columns of its submatrix $X = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix}$ satisfy all quadratic equations in (1).

**Definition 1** *A sensor network is* uniquely localizable *if there is a unique $X \in \mathbb{R}^{d\times n}$ whose columns satisfy (1), and there is no $\bar{X} \in \mathbb{R}^{h\times n}$, for $h > d$, whose columns satisfy (1) and $\bar{X} \neq (X; \mathbf{0})$. In other words, there is no non-trivial extension of $X \in \mathbb{R}^{d\times n}$ into higher dimension $h > d$ that also satisfies (1) [30].*

Note that the notion of unique localizability is stronger than the notion of global rigidity. A sensor network is globally rigid only if there is a unique $X \in \mathbb{R}^{d\times n}$ that satisfies (1), but it may also have a solution in a higher dimension space, that is a non-trivial extension of $X \in \mathbb{R}^{d\times n}$, which satisfies (1) [1, 30].

The following theorem was proved in [30]:

**Theorem 1** *An SNL problem instance is uniquely localizable if and only if the maximum rank solution of its SDP relaxation (2) has rank $d$, or equivalently, every solution matrix $Z$ of (2) satisfies $Y = X^T X$. Moreover, such a max-rank solution matrix can be computed approximately in polynomial time.*

The theorem asserts that the certification of a uniquely localizable network instance can be achieved by solving a convex optimization problem; the proof is constructive and produces a unique realization or localization solution for the original problem (1).

The dual of the SDP relaxation (2)

$$\text{minimize } I_d \bullet V + \sum_{(i,j)\in E} y_{ij} d_{ij}^2 + \sum_{(k,j)\in \bar{E}} w_{kj} \bar{d}_{kj}^2$$

$$\text{subject to } \begin{pmatrix} V & 0 \\ 0 & 0 \end{pmatrix} + \sum_{(i,j)\in E} y_{ij} A_{ij} + \sum_{(k,j)\in \bar{E}} w_{kj} \bar{A}_{kj} \succeq 0$$

is also useful, in that the solution to the dual tells us key properties about the primal. We define the dual slack matrix $U \in \mathbb{R}^{(d+n)\times(d+n)}$ as

$$U = \begin{pmatrix} V & 0 \\ 0 & 0 \end{pmatrix} + \sum_{(i,j)\in E} y_{ij} A_{ij} + \sum_{(k,j)\in \bar{E}} w_{kj} \bar{A}_{kj}.$$

for $V \in \mathbb{R}^{d\times d}$. The dual slack matrix $U$ is optimal if and only if it is feasible and meets the complementarity condition, $ZU = 0$. If complementarity holds, then $\text{rank}(Z) + \text{rank}(U) \leq (d+n)$, and since $\text{rank}(Z) \geq d$, this means that $\text{rank}(U) \leq n$. Thus, if an optimal dual slack matrix has rank $n$, then every solution to (2) has rank $d$ [30]. In fact, we have a stronger notion on localizability:

**Definition 2** *A sensor network is* strongly localizable *if there exists an optimal dual slack matrix with rank $n$.*

Again, such a max-rank dual solution matrix can be computed approximately in polynomial time using SDP interior-point algorithms.

## 1.2 Our Contributions

In this paper, we present new conditions that guarantee unique localizability of the SDP relaxation of the problem, i.e., conditions that ensure the SDP will give the correct solution so that the sensor network can be localized in polynomial time. We also enhance the relaxation such that the new SDP relaxation will produce a correct solution in dimension $d$ to satisfy (1), even when the standard SDP relaxation (2) may not. More precisely, our result is twofold:

1. A very popular graph in the context of sensor network localization is the unit-disk graph, where any two sensor points (or a sensor point and an anchor point) are connected if and only if their Euclidean distance is less than a given radio radius value $r$. It has been observed that when the radius (or radio range) increases, more sensors in the network can be correctly localized. There is an asymptotic analysis to explain this phenomenon when the sensor points are uniformly distributed in a unit-square [3]. In this paper, we present a *non-asymptotic* bound on the radius requirement of the points in order to ensure the network is uniquely localizable. Specifically, we decompose the area into sub-regions, which allows us to analyze the localizability of points in each sub-region, as opposed to the localizability of each individual point. We then determine the probability that all sensors will be localizable given a concentration of the sensors in a specified area. This may have practical impact by providing guidance on communication power ranges that ensure the network is uniquely localizable.

2. The basic SDP localization model (2) is an SDP feasibility problem. An open question has been to determine whether adding certain objective functions to the basic model improves localizability of the problem; that is, the SDP feasible region contains high rank solutions, but with certain objectives the SDP optimal solution is guaranteed to be unique and low-rank. We give an affirmative answer for a generic class of graph, by identifying an objective function that will always result in a correct localization for this class of graphs. Our result may also have an influence on Compressed Sensing, which uses an objective function to produce the sparsest solution. Based on this idea, we present numerical results by comparing several SDP objective functions to illustrate their effectiveness.

Moreover, although our theoretical analyses are based on exact distance measurements, similar extensions of our model (established in earlier SDP work) would be applicable to noisy distance data.

## 1.3 Paper Organization

The organization of this paper is as follows. First, Section 2 derives a lower bound for the connectivity radius in a sensor network that guarantees unique localizability with high probability. In section 3, we prove that given a triangulation (i.e., a planar, chordal and convex) graph, if the sum of the distances between nodes that do *not* have an edge between them is maximized, then the graph will be strongly localizable. We use this idea, and test a number of heuristic objective functions on a large number of random sensor networks to determine how well each works in practice. Our results for these heuristics are presented in Section 4.

4

# 2   Bounding the Connectivity Radius

In this section, we consider the unit-disk graph model [7, 8, 14] for sensor networks, where the Euclidean distance between any two sensor points (or a sensor point and an anchor point) is known (and the two points are connected) if and only if the distance between them is less than a given connectivity radius $r$. Assuming that the sensor points are randomly distributed in a region, we then establish a lower bound on radius $r$ that guarantees unique localizability, with high probability, of the sensor network formed based on radius $r$. Actually, we establish a lower bound on radius $r$ to ensure that the unit-disk graph is a $(d+1)$-lateration graph, which would be sufficient for unique localizability.

**Definition 3** *For some $d, n \geq 1$, the graph $G(V, E)$ is a $(d+1)$-lateration graph if there exists a permutation of the points, $\{\pi(1), \pi(2), \ldots, \pi(n)\}$, such that the edges of the sub-graph $\pi(1), \ldots, \pi(d+1)$ form a complete graph, and each successive point $\pi(j)$ for $j \geq d+2$ is connected to $d+1$ points in the set $\{\pi(1), \ldots, \pi(j-1)\}$. This permutation of the points, $\pi$, is called a $(d+1)$-lateration ordering.*

It is shown in [34] that if a sensor network graph contains a spanning $(d+1)$-lateration graph and the points are in general position, then it is uniquely localizable. Zhu et al. [34] provide a rigorous proof, which is based on the intuitive concept that a complete graph of $d+1$ points in general position can be always uniquely localized, and any point connected to $d+1$ uniquely localized points can be also uniquely localized.

Define $r(p)$ to be the connectivity radius of the randomly distributed sensor points that ensures the network is uniquely localizable with probability at least $p$ when the disk radius is $r(p)$. To find a lower bound on $r(p)$, we can find a connectivity radius for which the unit-disk graph $G(V, E)$ will contain a spanning $(d+1)$-lateration graph with at least probability $p$.

We approach the problem by considering a unit hypercube, $\mathcal{H} = [0, 1]^d$, which contains all the sensor points. We then split the region $\mathcal{H}$ into a grid of $M$ equal sub-hypercubes in dimension $d$, say $h_1, h_2, \ldots, h_M \subset \mathcal{H}$, where each sub-hypercube $h_i$ will have a volume of $1/M$, and the length of each of its edges will be $\ell := 1/\sqrt[d]{M}$. Without loss of generality, we can assume $M = b^d$, where $b$ is a positive integer and $b \geq 3$. Similarly, if the region considered is a hyper-rectangle in dimension $d$, we can assume $M = b_1 \cdot b_2 \cdots b_d$, where $b_i \geq 3$ for $i = 1, \ldots, d$ are positive integers. This partition will allow us to analyze the localizability of sub-hypercubes in the region and their containing points, as opposed to analyzing the localizability of each point individually.

## 2.1   Ensuring a Clique in the Graph

Since a $(d+1)$-lateration ordering on the points will begin with a $(d+1)$-clique, we first find a lower bound on the radius $r$ to ensure there exists at least one clique of $d+1$ points in the graph.

**Proposition 1** *Let $\mathcal{H}$ contain $n$ points, and $r \geq \ell \sqrt{d} = \frac{\sqrt{d}}{\sqrt[d]{M}}$ and $M \leq \frac{n-1}{d}$ (or equivalently $r \geq \frac{\sqrt[d]{d}\sqrt{d}}{\sqrt[d]{n-1}}$). Then, there exists at least one clique of $d+1$ points in the unit-disk graph $G(V, E)$.*

**Proof:** Note that $\frac{\sqrt{d}}{\sqrt[d]{M}}$ is the length of the diagonal of each sub-hypercube $h_i$. Thus, if $r$ is lower bounded by the given value, then every point in a sub-hypercube will be connected to any other point in the same sub-hypercube. Furthermore, since there are at most $\frac{n-1}{d}$ sub-hypercubes,

5

by the pigeon-hole principle, at least one of them contains at least $(d + 1)$ points and they must form a clique of $d + 1$ points in the unit-disk graph with given radius $r$. $\square$

In what follows, we fix $n = d \cdot M + 1 = d \cdot b^d + 1$. We will initialize the spanning $(d + 1)$-lateration graph construction by choosing $r$ according to this lower bound, and let the points in the $(d + 1)$-clique be the first $d + 1$ points in the lateration ordering. Since these points are randomly distributed, they must be in general position with probability one. Thus, we may assume that these $d + 1$ points are anchors for the sensor network. This assumption is without loss of generality since our bound on the radius $r$ established in the following sections will be much greater than the bound specified in Proposition 1, simply because we need to ensure that not only does there exist a clique of $d + 1$ points, but also all sensor points in $\mathcal{H}$ form a spanning $(d + 1)$-lateration graph with a high probability.

## 2.2 Binomial Distribution Model

One way to let the sensor points be randomly distributed throughout the area of $\mathcal{H}$ is to let the points be binomially distributed throughout each sub-hypercube of $\mathcal{H}$. More specifically, the number of points, $Y_i$, placed in each sub-hypercube $h_i$, for $i = 1, ..., M$, will be independently and binomially generated according to $Y_i \sim B\left(n, \frac{1}{M}\right)$ with $n = d \cdot M + 1 = d \cdot b^d + 1$. Once $Y_i$ is generated, we let these $Y_i$ sensor points be arbitrarily placed in general position within sub-hypercube $h_i$.

Using this binomial distribution model, let $S_n = \sum_{i=1}^{M} Y_i$ denote the total number of points in the hypercube $\mathcal{H}$. Since values $Y_i$s are independently and identically distributed and all sub-hypercubes are equally sized, the total number of points will be more or less evenly distributed in the entire hypercube $\mathcal{H}$. Furthermore, by properties of the binomial distribution,

$$\mathrm{E}\left[S_n\right] = M \cdot \mathrm{E}\left[Y_1\right] = M\left(\frac{n}{M}\right) = n$$

$$\mathrm{Var}\left(S_n\right) = M \cdot \mathrm{Var}\left(Y_1\right) = M \cdot \left[\frac{n}{M}\left(1 - \frac{1}{M}\right)\right] = n\left(1 - \frac{1}{M}\right).$$

Thus, $\frac{S_n}{n} \to 1$ almost surely and the assumption of binomially distributed sensor points throughout each sub-hypercube is statistically equivalent to assuming a uniform distribution of $n$ points throughout the whole region $\mathcal{H}$ when $M$ is sufficiently large.

## 2.3 Connectivity Bound

We now form further conditions on the connectivity radius $r$ to ensure that the unit-disk graph $G$ contains a spanning $(d + 1)$-lateration graph. We have assumed that the points are binomially distributed in each sub-hypercube, parametrized as $B\left(n, \frac{1}{M}\right)$. First, $r$ must satisfy Proposition 1, since it ensures a $(d+1)$-clique in $G$. These points in the clique will represent the first $d + 1$ points in the lateration ordering $\pi$ of a spanning $(d + 1)$-lateration graph (Definition 3).

We construct an improved bound on the probability of localizability through an ordering of the hypercubes, $h_i \in \mathcal{H}$, and hence an ordering on the points. For simplicity, we prove the following lemmas for the case of $d = 2$, and we refer to the sub-hypercubes as sub-squares. We also refer to $(d + 1)$-lateration when $d = 2$ as *trilateration*. However, we note that the same analysis can be applied to hypercubes in higher dimensions, and our bound $r \geq 2\ell\sqrt{2}$ in Lemmas 1–3 is analogous to the bound $r \geq 2\ell\sqrt{d}$ in dimension $d$.

**Lemma 1** *Assume that each sub-square in $\mathcal{H} \in \mathbb{R}^2$ has at least one point, and $r \geq 2\ell\sqrt{2}$. If the points of three sub-squares in the same row in three consecutive columns are in the trilateration ordering, then the points in all sub-squares in those three columns are also in the ordering. Similarly, if the points in three consecutive sub-squares in the same column are in the trilateration ordering, then the points in all sub-squares in those three columns are also in the ordering.*

**Proof:** First, note that the lower bound $r \geq 2\ell\sqrt{2}$ ensures that all points in a given sub-region are connected to all points in a neighboring sub-region, that is, all points in a sub-region that shares either an edge or a point.

For ease of explanation, consider the case that all points in the first three sub-squares in the first row of the grid are already in the trilateration. Since all points in the second row of the grid in the first three columns are within the connectivity range of all three points in the first row, these points are in the trilateration. Similarly, since each sub-square has at least one point, all points in the third row of the grid in the first three columns are within the connectivity range of three points in the second row, so these points are also in the trilateration. This pattern continues, until all points in the first three columns of the grid are in the trilateration.

A generalization of this shows that if there are three sub-squares in the same row and in consecutive columns with points in the trilateration, and each sub-square has at least one point, then all points in the corresponding columns are also in the trilateration.

An analogous result holds for three sub-squares in the same column and in consecutive rows. □

Lemma 1 states that if there are three sub-squares in a row with points in the trilateration, then the trilateration ordering extends to all squares in the corresponding columns. This concept is used below in Lemma 2, which analyzes the cases depicted in Figure 1.
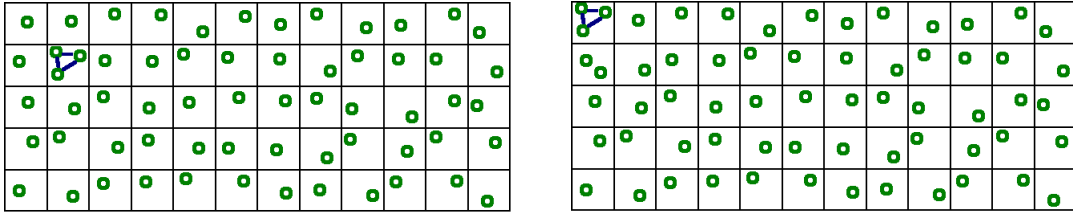


Figure 1: Conditions as described in Lemma 2 to ensure trilateration

**Lemma 2** *Assume there is at least one point in each sub-square and $r \geq 2\ell\sqrt{2}$. Then the associated unit-disk graph contains a spanning trilateration graph if either:*

a) *There is a 3-clique in a non-corner sub-square*

b) *There is a 3-clique in a corner sub-square and one of its neighbor squares has at least two points*

**Proof:** Again, note that $r \geq 2\ell\sqrt{2}$ ensures all points in a given sub-square are connected to all points in neighboring sub-squares. We show that if either of the conditions of Lemma 2 are satisfied, then there exists a trilateration ordering on the points in the graph.

a) Consider the example in the left grid of Figure 1, where there is a 3-clique in a non-corner sub-square. Let the points in this clique be the initial 3 points in the trilateration ordering. All

points in the sub-squares in the first three rows and first three columns are connected to this clique; let the points in these squares be next in the trilateration ordering.

By Lemma 1, all points in the first three rows of sub-squares are in the trilateration ordering. Since there are at least three columns in $\mathcal{H}$, the same argument applies for the columns, and inductively, there is a trilateration ordering on the points that spreads throughout the entire hyperspace $\mathcal{H}$.

b) Now consider the right grid of Figure 1, where there is a 3-clique in a corner sub-square, and there are at least two points in a neighboring sub-square. Let the points in this corner clique be the first 3 points in the trilateration ordering. All points in the three sub-squares that neighbor the 3-clique are connected to the clique and hence in the trilateration ordering. Next, let points the sub-squares in the third row and first three columns be the succeeding points in the ordering, followed by the sub-squares in the third column and first three rows. Using a similar argument as before from Lemma 1, we can construct a trilateration on the points in the graph, and all points are in the trilateration.

Therefore, if the conditions of Lemma 2 hold, the associated unit-disk graph contains a spanning trilateration graph.  □

The above lemma provides sufficient, but not necessary, conditions on a network for trilateration to exist, which implies unique localizability. Moreover, these are strict conditions for a sensor network, since the distribution of sensors in a network may not always ensure that there is one sensor in each sub-square. Thus, we extend these conditions to a more general case, and allow for the possibility of empty sub-squares. Clearly, too many empty sub-squares will result in a graph that is not uniquely localizable; also, if empty sub-squares exist, there must be restricting conditions to ensure the graph is not too sparse to ensure localizability. Thus, we establish additional properties of the graph that ensure a trilateration but allow for empty sub-squares.

**Definition 4** *Two neighboring sub-squares are called* adjacent *neighbors if they do not share any edges, but share a point; neighbors that share an edge are called* simple *neighbors. A sub-square is called* densely surrounded *if all its simple neighbors have at least two points and one of the simple neighbors has at least 3 points.*
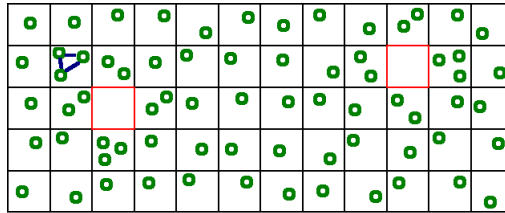


Figure 2: Example of a grid that satisfies Lemma 3

**Lemma 3** *Assume every empty sub-square is densely surrounded and $r \geq 2\ell\sqrt{2}$. Then the associated unit-disk graph contains a spanning trilateration graph if there is a 3-clique in a non-corner sub-square.*

8

**Proof:** Consider the grid in Figure 2, which shows an example of a non-corner 3-clique, and densely surrounded empty sub-squares. Notice that if an empty sub-square is densely surrounded, a trilateration ordering will spread around the empty sub-square: if the trilateration starts on one side of the empty sub-square, there are enough points on all sides of the empty sub-square for the ordering to continue around an empty sub-square.

This example illustrates that empty sub-squares do not necessarily interfere with the trilateration ordering under the given conditions. Notice that continuing the ordering among the first three rows of the grid, starting at the first column and adding points consecutively in the first three rows of the second, third and fourth columns will result in a trilateration ordering.

Therefore, if the condition of Lemma 3 holds, the associated graph contains a spanning trilateration graph, and hence is uniquely localizable in dimension 2. □

We now use the fact that a sensor network containing a spanning trilateration is uniquely localizable [34] to establish a lower bound on the probability that the unit disk sensor network with radius $r \geq 2\ell\sqrt{2}$ is localizable. Define the two events:

$$C := \{\text{There are only 3-cliques in corner sub-squares}\},$$

$$\widehat{C} := \{\text{There is a 3-clique in a non-corner sub-square}\}.$$

Then, the probability that a graph with such randomly distributed points is uniquely localizable will be

$$\mathrm{P}\{\text{uniquely localizable}\} = \mathrm{P}\left\{\text{uniquely localizable}|\widehat{C}\right\}\mathrm{P}\left\{\widehat{C}\right\} + \mathrm{P}\{\text{uniquely localizable}|C\}\mathrm{P}\{C\}$$

$$\geq \mathrm{P}\left\{\text{uniquely localizable}|\widehat{C}\right\}\mathrm{P}\left\{\widehat{C}\right\}$$

Given that the total number of sub-squares is $M = b^2$ (for some integer $b \geq 3$), we introduce a parameter $\alpha := \sqrt{\frac{n}{M}}$ (or $\alpha := \sqrt[d]{\frac{n}{M}}$ for general $d$) such that $\ell = \alpha/\sqrt{n}$ is the edge-length of each sub-square and we can use the same connectivity radius lower bound as before, now in terms of $\alpha$, $r(\alpha) \geq (2\alpha\sqrt{2})/\sqrt{n}$. The distribution of point number in each sub-square is binomial $B\left(n, \frac{1}{M}\right)$, and there are a total of $(M - 4)$ non-corner sub-squares in $\mathcal{H}$. Thus, the probability that there is a 3-clique in a non-corner sub-square is

$$\mathrm{P}\left\{\widehat{C}\right\} = 1 - \left(\sum_{i=0}^{2}\binom{n}{i}\left(\frac{1}{M}\right)^i\left(1 - \frac{1}{M}\right)^{n-i}\right)^{M-4}.$$

Let $k$ be the number of empty sub-squares. By Lemma 2, $\mathrm{P}\left\{\text{uniquely localizable}|k = 0, \widehat{C}\right\} = 1$, and if $p_0 = (1 - \frac{1}{M})^n$ is the probability that one specific sub-square is empty, we have

$$\mathrm{P}\{k = i\} = \binom{M}{i}p_0^i(1 - p_0)^{M-i}.$$

Moreover, for any $i < M - 4$, we have

$$\mathrm{P}\left\{\widehat{C}|k = i\right\} \geq 1 - \left(\sum_{j=0}^{2}\binom{n}{j}\left(\frac{1}{M}\right)^j\left(1 - \frac{1}{M}\right)^{n-j}\right)^{M-4-i} := p_{\widehat{C},i}.$$

9

From Lemma 3, we know

$$P\left\{\text{uniquely localizable}|k = i, \widehat{C}\right\} \geq P\left\{\text{empty sub-squares are densely surrounded}|k = i, \widehat{C}\right\}.$$

The conditions of Lemma 3 require that empty sub-squares not have empty simple neighbors; thus we first find the probability that a sub-square does not have empty simple neighbors. Assume there are $k$ empty sub-squares, say $s_1, s_2, \ldots, s_k$. Because of the independence assumption, these empty sub-squares are uniformly distributed.

Given the empty sub-square $s_1$, the probability that $s_2$ is not a simple neighbor of $s_1$ is at least $\left(1 - \frac{4}{M-1}\right)$; the probability that $s_3$ is not a simple neighbor of $s_1$ or $s_2$ is at least $(1-2\cdot\frac{4}{M-2})$; and so on, so that the probability that no two empty sub-squares are neighbors is at least $\prod_{j=1}^{k-1}(1 - \frac{4j}{M-j})$. Moreover, the probability that an empty sub-square is densely surrounded, i.e., that all simple neighbors of an empty sub-square have at least two points and at least one of them has more than two points, is:

$$\hat{p} = P\left\{\text{All simple neighbors have at least two points}\right\} - P\left\{\text{All simple neighbors have exactly two points}\right\}$$

$$= \left[1 - \sum_{j=0}^{1}\binom{n}{j}\left(\frac{1}{M}\right)^{j}\left(1 - \frac{1}{M}\right)^{n-j}\right]^{4} - \left[\binom{n}{2}\left(\frac{1}{M}\right)^{2}\left(1 - \frac{1}{M}\right)^{n-2}\right]^{4}$$

Thus, the probability that all empty sub-squares are densely surrounded is

$$P\left\{\text{empty sub-squares are densely surrounded}|\ k = i, \widehat{C}\right\} \geq \hat{p}^{i} \cdot \prod_{j=1}^{i-1}(1 - \frac{4j}{M - j}).$$

Note that the right hand side of the above equation is positive if $i < M/5$. Thus, we only consider grids with less than $u := \lfloor M/5 \rfloor - 1$ empty squares. Finally, we have the lower bound given by the following expression:

$$P\left\{\text{uniquely localizable}\right\}$$

$$\geq P\left\{\text{uniquely localizable}|\widehat{C}\right\} P\left\{\widehat{C}\right\}$$

$$= \sum_{i=0}^{u} P\{\text{uniquely localizable}|k = i, \widehat{C}\} P\{\widehat{C}|k = i\} P\left\{k = i\right\}$$

$$\geq \sum_{i=0}^{u} P\{\text{uniquely localizable}|k = i, \widehat{C}\} P\left\{k = i\right\} p_{\widehat{C},i}$$

$$\geq p_{\widehat{C},0} P\left\{k = 0\right\} + \sum_{i=1}^{u} p_{\widehat{C},i} P\left\{k = i\right\} \times P\left\{\text{empty sub-squares are densely surrounded}|k = i, \widehat{C}\right\}$$

$$\geq p_{\widehat{C},0} P\left\{k = 0\right\} + \sum_{i=1}^{u} \hat{p}^{i} \cdot p_{\widehat{C},i} P\left\{k = i\right\} \times \prod_{j=1}^{i-1}(1 - \frac{4j}{M-j}). \tag{3}$$

For different values of $n$ (viewed as the total number of sensor points), we can find values of $M$, and thus $\alpha$ (where $\alpha^2$ can be viewed as the average number of sensor points in each sub-square), such that the right hand side of Equation (3) is at least 0.99. Figures 3a and 3b show $\alpha$ and $r$ versus the number of points $n$ such that the right hand side of Equation 3 is at least 0.99.

We also depicted our connectivity bound against two other bounds in Figure 4. One can see that our and Aspnes' bounds are almost identical for any value of $n$. Thus, our result shows that the bound of Aspnes et al. in [3] (of $r > \frac{2\sqrt{2}\sqrt{\log n}}{\sqrt{n}}$ for $d = 2$) is true even when $n$ is small, although it was initially proved to be an asymptotic bound when $n$ is sufficiently large. Note that our bound,
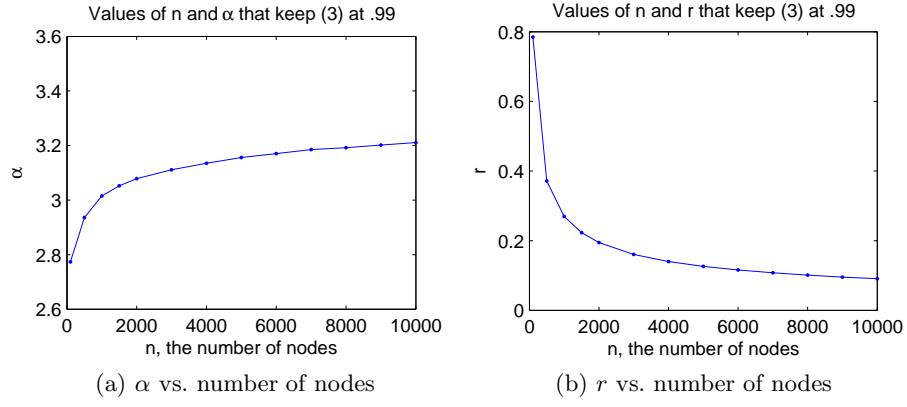
(a) $\alpha$ vs. number of nodes

(b) $r$ vs. number of nodes

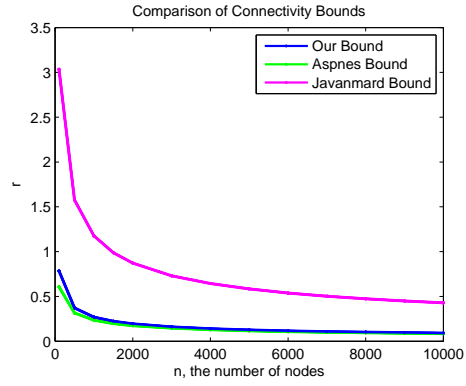Figure 3: Bound on the Connectivity Radius



Figure 4: Comparison of our connectivity bound to those of Javanmard [23] and Aspnes [3]

11

while not in an analytical form, is proved for any value of $n$. On the other hand, we recently learned of another asymptotic bound that was independently developed by Javanmard and Montanari [23]. As seen from Figure 4, their bound is actually weaker than ours and Aspnes'.

Our connectivity result was proved for $\mathcal{H} = [0, 1]^2$, i.e., the unit square in dimension 2. The result can be extended to dimension $d > 2$. In summary, we have

**Theorem 2** *Let $\mathcal{H} \in [0, 1]^d$ be the unit hypercube in dimension $d$ and be partitioned into a grid of $M = b^d$ equal sub-hypercubes, say $h_1, h_2, \ldots, h_M \subset \mathcal{H}$, where $\ell = 1/b$ is the edge length of each sub-hypercube. Let the number of sensor points in each sub-hypercube be independently and binomially generated according to $B\left(n, \frac{1}{M}\right)$ where $n = d \cdot M + 1$, and let one of the sub-hypercubes contain $d + 1$ anchors. Then, if the connectivity radius satisfies $r \geq 2\ell\sqrt{d}$, the probability that the sensor network is uniquely localizable is given by expression (3).*

Again, the parameter $n$ of the binomial distribution can be viewed as the total number of sensor points in the region. We can also extend our result to another region $\mathcal{H}$ in dimension $d$ into a grid of $M$ equal sub-hypercubes in dimension $d$, say $h_1, h_2, \ldots, h_M \subset \mathcal{H}$, where each sub-hypercube $h_i$ will have a volume of $1/M$, and the length of each of its edges will be $\ell := 1/\sqrt[d]{M}$. For example, we can assume $M = b_1 \cdot b_2 \cdots b_d$, where $b_i \geq 3$ for $i = 1, \ldots, d$ are positive integers.

# 3   Unique Localization of Triangulation Graph

The basic SDP localization model (2) is an SDP feasibility problem. When the network is not uniquely localizable, the max-rank of SDP feasible solutions is strictly greater than $d$. In practice, one may still be interested in finding a feasible SDP solution with rank $d$, representing one possible localization of points in $\mathbb{R}^d$. In this section, we show that adding an objective function that maximizes the sum of certain distances in a triangulation graph (in $\mathbb{R}^2$) will produce a rank-2 SDP solution. The result should be applicable to $d > 2$.

**Definition 5** *Consider a set of points $\mathcal{P} = \{p_1, p_2 \ldots p_n\} \in \mathbb{R}^2$. A* triangulation, *$\mathcal{T}_\mathcal{P}$, of the points in $\mathcal{P}$ is a subdivision of the convex hull of $\mathcal{P}$ into simplices (triangles) such that the edges of two simplices do not intersect or share a common face.*

Triangulation graphs and their properties have been studied in the literature [5, 12, 26]. Bruck et al. [12] showed that an embedding on a unit disk graph with local angle information (angles between points) is NP-hard, while the same problem on a triangulation graph is not. Araujo et al. [5] introduced an algorithm to construct a triangulation graph from a unit disk graph with $\mathcal{O}(n \log n)$ bit communications between points.

**Definition 6** *For a triangulation $\mathcal{T}_\mathcal{P}$, we define a* triangulation graph *$G_{\mathcal{T}_\mathcal{P}}(V, E)$ such that $V = \mathcal{P}$ and $(p_i, p_j) \in E$ if and only if $(p_i, p_j)$ is an edge of a simplex in $\mathcal{T}_\mathcal{P}$.*

We formally decompose a triangulation $\mathcal{T}_\mathcal{P}$ into an initial clique $K_3$ and a set of actions $\mathcal{A} = \{a_1, a_2 \ldots a_m\}$, where an action $a_i$ consists of adding a point and connecting it to either two adjacent points or two connected external points, where a point is called *external* if it is not strictly inside the convex hull of a cycle in the graph. This leads us to the following lemma, whose proof is omitted.

(a) Recursive Construction of a Triangulation

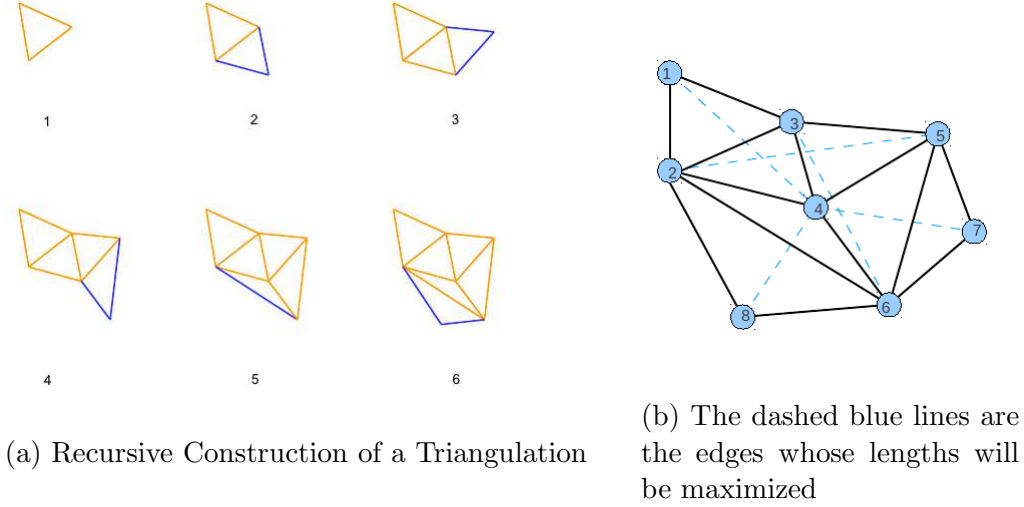(b) The dashed blue lines are the edges whose lengths will be maximized

Figure 5: Triangulation Examples

**Lemma 4** *A triangulation can be constructed recursively by either adding an external point that connects to two adjacent points of a simplex (triangle) already in $\mathcal{T}_\mathcal{P}$ such that the new edges do not cross any existing edges (see Figure 5a, 1-4), or simply connecting two external points already in $\mathcal{T}_\mathcal{P}$ to form a triangle (see Figure 5a, 5).*

**Proof:** By induction on an external point; see Figure 5a. □

**Definition 7** *In a triangulation graph,* adjacent triangles *are two triangles which share a common edge. A* virtual edge *exists between two points $i$ and $j$ when $i$ and $j$ belong to adjacent triangles, but $(i,j) \notin E$. The set of virtual edges between sensors is denoted $E_v$, and between sensors and anchors is denoted $\bar{E}_v$.*

**Theorem 3** *If we add an objective function to the SDP model (2) which maximizes the sum of the lengths of all virtual edges in a generic triangulation graph, then the rank of the optimal SDP solution is $d$ and it produces the correct localization.*

**Proof:** We prove this via induction by assuming it is true for any triangulation graph with $n$ points and then showing that it is also true for graphs with $n+1$ points. It is clearly true for a single simplex when $n = 3$.

Using the same notation as in (2), the primal SDP relaxation with this objective is:

$$
\begin{aligned}
\text{maximize} \quad & \sum_{(k,j)\in\bar{E}_v} \bar{A}_{kj} \bullet Z + \sum_{(i,j)\in E_v} A_{ij} \bullet Z \\
\text{subject to} \quad & Z_{(1:d,1:d)} = I_d \\
& A_{ij} \bullet Z = d_{i,j}^2, \forall (i,j) \in E \\
& \bar{A}_{kj} \bullet Z = \bar{d}_{k,j}^2, \forall (k,j) \in \bar{E} \\
& Z \succeq 0
\end{aligned}
\tag{4}
$$

and the dual of (4) is:

$$\text{minimize} \quad I \bullet V + \sum_{(i,j)\in E} y_{ij} d_{ij}^2 + \sum_{(k,j)\in \bar{E}} w_{kj} d_{kj}^2$$

$$\text{subject to} \quad U = \begin{pmatrix} V & 0 \\ 0 & 0 \end{pmatrix} + \sum_{(i,j)\in E} y_{ij} A_{ij} + \sum_{(k,j)\in \bar{E}} w_{kj} \bar{A}_{kj}$$

$$- \sum_{(k,j)\in \bar{E}_v} \bar{A}_{kj} - \sum_{(i,j)\in E_v} A_{ij} \tag{5}$$

$$U \succeq 0$$

Let $X^n \in \mathbb{R}^{d\times n}$ be the correct locations of points, where the superindex $n$ represents the number of points. By the induction assumption, the solution to (4) is $Z^n := \begin{pmatrix} I_d & X^n \\ (X^n)^T & (X^n)^T X^n \end{pmatrix}$. Moreover, the optimal dual slack matrix $U^n$ satisfies $U^n \bullet Z^n = 0$ and has rank $n$; we can write the optimal dual slack matrix in terms of its submatrices $U^n = \begin{pmatrix} U_{11}^n & U_{12}^n \\ U_{21}^n & U_{22}^n \end{pmatrix}$, where $U_{11}^n \in \mathbb{R}^{d\times d}$ and $U_{22}^n \in \mathbb{R}^{n\times n}$ and $U_{22}^n \succ 0$. Note that the complementarity condition $U^n \bullet Z^n = 0$ means the elements of $U^n$ represent a stress on each edge such that the total force at all non-anchor points is zero (assuming, without loss of generality, a stress of $-1$ on all virtual edges).

We decompose the triangulation graph into an initial simplex $K_3$, and actions $\mathcal{A} = \{a_1, a_2 \ldots a_m\}$. Without loss of generality, we assume the points in the first triangle are anchor points and let the last points added to the graph be $x_{n+1}$. For example, consider Figure 5(b); let $U^7$ be the dual slack matrix on points 1-7 and assume the subgraph induced on the first 7 points is uniquely localizable. When point 8 is added along with its incident edges, points $(2, 4, 6, 8)$ form a clique (when including the virtual edge between 4 and 8, which is unique when its length is maximized). Thus, the sub-graph induced on points $(2, 4, 6, 8)$ is uniquely localizable and its stress matrix is positive semi-definite with rank 1 [19].

Now consider the general case, where $x_{n+1}$ is the last point added to the graph. A new triangle is created by adding $x_{n+1}$, its adjacent triangle and the virtual edge, which forms a 4-clique. Define $\Omega_0$ to be the corresponding PSD stress matrix (with rank at least 1) on the graph formed by $x_{n+1}$, the two points adjacent to $x_{n+1}$ (say, $g$ and $h$) and the point with which $x_{n+1}$ has a virtual edge (say, $k$). We examine the case where $g$ and $h$ are sensors, however the case where at least one of them is an anchor is an easy extension. Assume $\Omega_0$ is normalized such that the stress associated with the virtual edge is $-1$.

$$\Omega_0 = \begin{pmatrix} -1 + y_{gk} + y_{hk} & -y_{gk} & -y_{hk} & 1 \\ -y_{gk} & y_{gk} + y_{gh} + y_{g,n+1} & -y_{gh} & -y_{g,n+1} \\ -y_{hk} & -y_{gh} & y_{hk} + y_{gh} + y_{h,n+1} & -y_{h,n+1} \\ 1 & -y_{g,n+1} & -y_{h,n+1} & -1 + y_{g,n+1} + y_{h,n+1} \end{pmatrix}$$
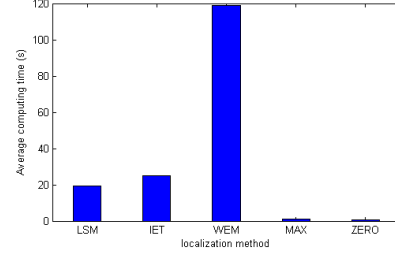
Note that $0 < (-1 + y_{g,n+1} + y_{h,n+1})$ is easily shown and consider the updated stress matrix with

$$U_{22}^{n+1} := \begin{pmatrix} U_{22}^n & 0_{n\times 1} \\ 0_{1\times n} & 0 \end{pmatrix} + \Omega$$

where $\Omega \in \mathbb{R}^{(n+1)\times(n+1)}$ is the stress matrix of the new edges, that is, $\Omega_{([g,h,k,n+1],[g,h,k,n+1])} = \Omega_0$.
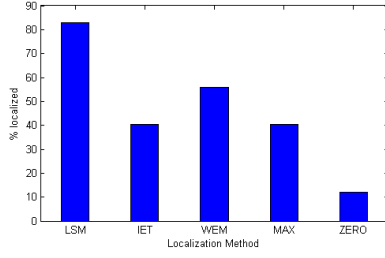
(a) Localization Results        (b) Time Results

The new matrix $U^{n+1}$ will be feasible for the dual, since the elements of $\Omega_0$ are constructed specifically so that the matrix $U^{n+1}$ will be of the right form, and $\Omega \succeq 0$, $U^n \succeq 0$ implies that $U^{n+1} \succeq 0$.

Define

$$Z^{n+1} := \begin{pmatrix} Z^n & \begin{pmatrix} x_{n+1} \\ (X^n)^T x_{n+1} \end{pmatrix} \\ \begin{pmatrix} x_{n+1}^T & x_{n+1}^T X_n \end{pmatrix} & x_{n+1}^T x_{n+1} \end{pmatrix}$$

as the correct locations of the updated points. Then,

$$U^{n+1} \bullet Z^{n+1} = U^n \bullet Z^n + \sum_{(i,j)} [\Omega_0]_{ij}(x_i^T x_j) = 0.$$

Moreover, we can show that $U_{22}^{n+1} \succ 0$. Assume not, i.e., that there is a vector $z \in \mathbb{R}^{n+1}$ such that

$$z^T U_{22}^{n+1} z = z^T \begin{bmatrix} U_{22}^n & 0 \\ 0 & 0 \end{bmatrix} z + z^T \Omega z = 0,$$

which only holds if $z^T \begin{bmatrix} U_{22}^n & 0 \\ 0 & 0 \end{bmatrix} z = 0$ and $z^T \Omega z = 0$. Since $U_{22}^n \succ 0$, this means that the first $n$ elements of $z$ are zero, i.e., $z_{(1:n)} = 0$. Thus,

$$z^T \Omega z = z_{n+1}^2 \Omega_{n+1} = z_{n+1}^2(-1 + y_{i,n+1} + y_{j,n+1}) = 0$$

which implies $z_{n+1} = 0$. Thus, $z^T U_{22}^{n+1} z = 0$ if and only if $z = 0$, implying $U_{22}^{n+1} \succ 0$ and $\text{rank}(U^{n+1}) = n + 1$. Therefore, $\text{rank}(Z^{n+1}) = d$ and the solution to (4) is exact.
$\square$

# 4   Heuristic Objective Function

Based on the finding of the early section, we tested a number of heuristic objective functions for the DP relaxation model on a large number of random sensor networks. The following objective functions and strategies are used:

1. (LSM) Initially maximize the sum of all non-edges, and do the following until either ($i$) the total error in edge lengths is sufficiently small, or ($ii$) the maximum number of iterations has been reached: (a) Randomly choose an edge and minimize its length in the objective function (while maximizing all others) (b) If this results in a smaller total error in edge lengths, then minimize this edge length in the objective function; otherwise, maximize it

15

2. (IET) For each non-edge, run two SDP localizations: (a) Maximize all edge lengths, (b) Maximize all edge lengths, except the length of the given edge, which is minimized. If the second method results in a small error, then minimize the length of this non-edge in the objective function; otherwise, maximize it.

3. (WEM) Run the SDP localization with no objective function to find a set of locations, $\hat{X}$. Then, find the edge $(i, j)$ with maximum error. Run the SDP localizations two more times, with the objective functions of maximizing and minimizing the length of this edge from either $i$ or $j$ to an anchor; choose to maximize (minimize) this distance in the objective function if maximizing (minimizing) the distance from $i$ or $j$ to an anchor resulted in less error.

4. (MAX) Maximize the sum of all the non-edge lengths

5. (ZERO) No objective function

The plots in Figures 6a and 6b show the results of each heuristic method. As the plot shows, (LSM) our-performed each other heuristic in terms of localization, however took much longer than (MAX) and (ZERO). The simple strategy of (MAX) seems to provide a good balance between the solution quality and time.

# References

[1] A. Y. Alfakih. On the universal rigidity of generic bar frameworks. *Contribution to Discrete Mathematics*, 5(3):7–17, 2010.

[2] A. Y. Alfakih, A. Khandani, and H. Wolkowicz. Solving euclidean distance matrix completion problems via semidefinite programming. In *Computational Optimization and Applications*, volume 12, pages 13–30, 1999.

[3] D. Angluin, J. Aspnes, M. Chan, M. J. Fischer, H. Jiang, and R. Peralta. Stably computable properties of network graphs. In V. K. Prasanna, S. Iyengar, P. Spirakis, and M. Welsh, editors, *Distributed Computing in Sensor Systems: First IEEE International Conference, DCOSS 2005, Marina del Rey, CA, USE, June/July, 2005, Proceedings*, volume 3560 of *Lecture Notes in Computer Science*, pages 63–74. Springer-Verlag, June 2005.

[4] D. Angluin, J. Aspnes, Z. Diamadi, M. J. Fischer, and R. Peralta. Computation in networks of passively mobile finite-state sensors. *Distributed Computing*, pages 235–253, March 2006.

[5] F. Arajo and L. Rodrigues. Fast localized delaunay triangulation. volume 3544, pages 81–93. Principles of Distributed Systems, 2005.

[6] J. Aspnes, T. Eren, D. K. Goldenberg, A. S. Morse, W. Whiteley, Y. R. Yang, B. D. O. Anderson, and P. N. Belhumeur. A theory of network localization. *IEEE Transactions on Mobile Computing*, 5(12):1663–1678, Dec 2006.

[7] J. Aspnes, D. Goldenberg, and Y. R. Yang. On the computational complexity of sensor network localization. In *Algorithmic Aspects of Wireless Sensor Networks: First International Workshop*, volume 3121 of *Lecture Notes in Computer Science*, pages 32–44. Springer-Verlag, 2004.

[8] M. Badoiu, E. D. Demaine, M. Hajiaghayi, and P. Indyk. Low-dimensional embedding with extra information. *Discrete and Computational Geometry*, 36(4):609–632, 2006.

[9] M. Belk and R. Connelly. Realizability of graphs. In *Discrete and Computational Geometry*, volume 37, pages 7125–7137, 2007.

[10] P. Biswas, T. Lian, T. Wang, and Y. Ye. Semidefinite programming based algorithms for sensor network localization. In *IPSN*, pages 46–54. ACM Press, April 2004.

[11] P. Biswas and Y. Ye. Semidefinite programming for ad hoc wireless network localization. In *IPSN*, pages 46–54, 2004.

[12] J. Bruck, J. Gao, and A. Jiang. Localization and routing in sensor networks by local angle information. pages 181–192. SIGMOBILE, 2005.

[13] N. Bulusu, J. Heidemann, and D. Estrin. Gps-less low-cost outdoor localization for very small devices. *IEEE Personal Communications*, 7(5):28–34, 2000.

[14] B. N. Clark, C. J. Colbourn, and D. S. Johnson. Unit disk graphs. *Discrete Mathematics*, 86(1–3):165–177, 1991.

[15] R. Connelly. Generic global rigidity. *Discrete and Computational Geometry*, 33(4):549–563, 2005.

[16] G. M. Crippen and T. F. Havel. Distance geometry and molecular conformation. In *Chemometrics Series, Research Studies Press Ltd., Taunton, Somerset, England*, volume 15, 1988.

[17] L. Doherty, K. S. J. Pister, and L. El Ghaoui. Convex position estimation in wireless sensor networks. In *IEEE INFOCOM*, volume 3, pages 1655–1663, 2001.

[18] T. Eren, E.K. Goldenber, W. Whiteley, and Y.R. Yang. Rigidity, computation, and randomization in network localization. In *IEEE INFOCOM*, volume 4, pages 2673–2684, 2004.

[19] S. J. Gortler, A. D. Healy, and D. P Thurston. Characterizing generic global rigidity. In *American Journal of Mathematics*, 2010.

[20] B. Hendrickson. Conditions for unique graph realizations. *SIAM Journal on Computing*, 21(1):65–84, 1992.

[21] J. Gao J. Bruck and A. A. Jiang. Localization and routing in sensor networks by local angle information. *ACM Transactions on Sensor Networks*, 5(1):181–192, 2009.

[22] B. Jackson and T. Jordan. Connected rigidity matroids and unique realizations of graphs. *Journal of Combinatorial Theory, Series B,*, 94(1):1–29, 2005.

[23] Adel Javanmard and Andrea Montanari. Localization from incomplete noisy distance measurements. 2011. `http://arxiv.org/abs/1103.1417v3`.

[24] Nathan Krislock and Henry Wolkowicz. Explicit Sensor Network Localization Using Semidefinite Representations and Clique Reductions. *SIAM Journal on Optimization*, 20(5):2679–2708, 2010.

[25] X. Li. Wireless ad hoc and sensor networks: Theory and applications. In *Cambridge University Press, New York*, 2008.

[26] X-Y. Li, G. Calinescu, P.-J. Wan, and Y. Wang. Localized delaunay triangulation with application in ad hoc wireless networks. *IEEE Trans. on parallel and distributed sys.*, 14(9), 2003.

[27] N.B. Priyantah, H. Balakrishnana, E.D. Demaine, and S. Teller. Mobile-assisted localization in wireless sensor networks. In *IEEE INFOCOM*, volume 1, pages 172–183, 2005.

[28] A. Savvides, C-C. Han, and M. B. Strivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *MobiCom*, pages 166–179, July 2001.

[29] A. M-C So and Y. Ye. A semidefinite programming approach to tensegrity theory and realizability of graphs. In *Symposium on Discrete Algorithms*, pages 766–775, 2006.

[30] A. M-C So and Yinyu Ye. Theory of semidefinite programming for sensor network localization. *Symposium on Discrete Algorithms*, pages 405–414, 2005.

[31] P. Tseng. Second-order cone programming relaxation of sensor network localization. *SIAM Journal on Optimization*, 18(1):156–185, 2007.

[32] Z. Wang, S. Zheng, Y. Ye, and S. Boyd. Further relaxations of the semidefinite programming approach to sensor network localization. *SIAM Journal on Optimization*, 19:655–673, 2008.

[33] Z. Yang, Y. Liu, and X-Y. Li. Beyond trilateration: On the localizability of wireless ad-hoc networks. In *IEEE INFOCOM*, pages 2392–2400, 2009.

[34] Z. Zhu, A. M-C So, and Yinyu Ye. Universal rigidity: Towards accurate and efficient localization of wireless networks. In *IEEE INFOCOM*, 2010.